Introduction

Our project, under the guidance of Barbara Wallace and the sponsorship of Bank One explored the potential of two technologies-J2EE and .NET-to handle the tasks of distributed programming. The research problem we faced was to determine which of the two technologies is better suited for a particular business problem. The team was presented with an application that Bank One is currently using, their Phone Book system, and the team's goal was to develop it in both technologies, create a criteria grid which compared the two technologies and apply the insight they gained from developing the software to populate the criteria grid. The purpose of that grid is to facilitate a future software development problem or a software migration problem, in which the bank would need to migrate a currently existing legacy system to one of the two environments.

Background

Currently, Bank One has numerous applications that have been developed in earlier technologies such as ASP. With the advance of Web Services, however, it became possible for a company to have both systems and use Web Services for any inter-system communication, i.e. from a J2EE module to call a .NET mid-tier module or vice versa. For many companies, Web Services shifted the focus of developing a system and migrating a currently existing system from "which technology to commit to" to "which technology to use for what particular application". Bank One saw the benefits of the flexibility of using both technologies and choosing one depending on the specific needs of an application. The problem that Bank One was now faced was how to decide on which of the two technologies to use given an application.

The two technologies that offer a solution to the problems faced by a distributed system are Sun's J2EE and Microsoft's .NET. The J2EE has had a longer presence on the market and is viewed by many as the more evolved environment. The J2EE is not bound to a particular operating system. As a result, components can be developed independently of the operating system and the hardware while still operating uniformly and making the platform differences transparent to a developer. The J2EE also offers a great variety of tools-IDE-s, build tools, application servers, Web Service and bean containers-which gives the additional flexibility to developers to use the tools that fit the task at hand best. An obvious limitation of the J2EE is that it only supports the Java programming language and developers are hence restricted.

In contrast, the .NET is a relative new-comer to the field, but it is powered by the momentum is steadily increasing its market share. It offers as many as 22 programming languages to developers but restricts them to using only the Windows OS. The .NET offers a single development environment and set of tools which developers can use and integrate with a relative ease.

Many attempts have been made to make the comparison of the two environments. What made our IPRO unique was that it took a more practical approach by learning the .NET and the J2EE, first, and then developed a single system in both in order to try and capture the learning curve involved in the process. The value that Bank One saw in this approach was that it would enable them to extrapolate from our learning curve and the problems we faced on each criterion to the potential learning curve of their employees and problems that they will be facing in the future, while taking one of the two routes.

The value of the decision of choosing one technology over the other is really hard to quantify because of the many implications involved in the process and the assumptions that have to be made. But on a higher level, what technology a company chooses to develop a system in or migrate a system to, really determines the future of that system for years to come and taking the wrong route could have potentially enormous, long-term negative consequences.

Research And Development Methodology

Our IPRO team consisting of 1 Ph D student, 3 graduate students and 9 undergraduate students from diverse majors like Design, Computer Science and Computer Engineering were introduced to a project where the main challenge was lack of knowledge about the technology being asked to employ - JAVA using the Struts architecture and .Net using the Active Server Pages framework. Our research into these two technologies began by each team member finding articles via the Web, describing advantages of one technology over the other. This gave us a fairly good idea of what each technology was capable of and their limitations as well.

The next step we took was dividing our team of 13 members into sub teams:

- Web Team to maintain the IPRO website,
- J2EE Team to create the application prototype using Struts,
- .Net Team to create the same application using ASP and
- Analysis Team to collect observations by the J2EE and ASP team on a list of approved criteria by our instructor, Barbara Wallace and Bank One.

Each person was involved in at least two or more teams. We also decided to use Yahoo Groups as our mode of communication within the team. All approved files were also uploaded to the group so any team member could access it. Every Sunday by midnight, each team sent status reports to the Web Team who uploaded the report to the IPRO website.

To aid us further in our project, we met with the Bank One employees 6 times at Bank One. At each meeting, we had a prepared agenda to discuss and clarify our questions or approve a certain document.

We followed the Waterfall Model for our software development. The Waterfall Model proceeds in 5 stages:

- Requirements Specification
- Design Specification
- Coding
- Test Specification
- Final Delivery

Consequently, our first document was the Requirements Specification. We were shown the current phonebook being used at Bank One at our second site meeting. The J2EE and .Net teams then worked on the requirements team-wise. Points from both teams were discussed before putting it into paper. After approval from Barb, we went on with discussing the document with Bank One. Key functionalities decided to be implemented were:

- Basic Search and Advanced Search in J2EE and .Net technologies,
- Basic search in J2EE via .Net web service and
- Basic search in .Net via j2EE web services

Web Services is a facility enabling an application in J2EE to communicate with .Net and vice versa. Following this, we began working on the Design Document. The J2EEand .NET teams produced a Design Document each describing the necessary classes and installations needed to create their respective application. These documents were also approved by Bank One before enforcing them in our coding. Simultaneously the User research group conducted the user interface testing to make sure the designed user interface is user friendly. Coding took approximately 4 weeks. Both teams put in considerable effort to code the application and then add Web Services to it. At the same time, black box test cases were developed by the Test Team. Black box testing was done to ensure that both applications functioned correctly and similarly. Simultaneously, the analysis team developed a list of criteria which coders in both teams observed as coding went along.

After both applications were coded, the test team verified that the test cases developed in the Test Spec returned the same results in the applications. Finally, both teams discussed their ratings for the criteria developed by the Analysis Team, comparing how easy/hard it was for one team to implement a certain criteria as opposed to the other.

RESULTS

The IPRO 337 team was successful in creating the Bank One phone book application in both the .NET/ C# and J2EE- Struts environment. Due to technical difficulty and a remote server we were unable to deploy the webservice application to the server. However we had the entire application both environments inclusive of webservices setup and running on local machines.

Please refer to the analysis report for details.

CONCLUSION AND RECOMMENDATION

From the research, application development, observations and analysis the IPRO team learnt that both the .NET/ C# and J2EE- Struts were at par with each other. But we inferred that the use of J2EE- Struts was more convenient for this application because of the ease of implementing frames for the user interface which was a requirement of the user interface group.

Due to the vastness of the technologies it was a tough learning curve for the team of beginners. But we learnt that the choice of which technology to choose depends on the requirements and design specifications of the applications.

Acknowledgements

We would like to acknowledge the following people for assisting us throughout the length of the IPRO:

- Barbara Wallace, the instructor for our IPRO. She guided the team through the software process and helped us keep on track with the documents and coding.
- Dan Ferguson, who assisted us with tips on how to meet the IPRO office requirements and relayed past IPRO experiences which benefited the team.
- Geoffrey Sutter, Bank One in-charge for the IPRO. He assisted our team by arranging the meetings and involving other Bank One employees with the project.
- Richard Douglas and Shikha Desai, from the Bank One user research team. They both assisted our IPRO user research team in conducting the usability test for our application's user interface.

The team used the following reference:

.NET Team

- MSDN library
- .NET Web Services Solutions, By: Kris Jamsa. ISBN: 0782141722
- ASP.NET: The Complete Reference, By: Matthew MacDonald. ISBN: 0072195134
- Mastering ASP.NET with C#, By: A. Russell Jones. ISBN: 0782129897
- Building XML Web Services for the Microsoft.NET Platform, By: Scott Short. ISBN: 0735614067
- www.microsoft.com
- http://www.gotdotnet.com/Community/UserSamples/
- http://www.csharphelp.com

J2EE Team

- Struts Kick Start by James Turner and Kevin Bedell. Sams Publications
- Servlets and JavaServer Pages 2nd Edition by Martin Hall and Larry Brown, Prentice Hall Pubs
- Programming Jakarta Struts by Chuck Cavaness O'Reilly pubs
- Web sphere help
- Mastering Jakarta Struts
- Different Java APIs

- 'Enterprise Java Beans Component Architecture' by Gail Anderson and Paul Anderson
- Mastering Enterprise JavaBeans, Second Edition
- JSP: The complete Reference

Documents Enclosed

Abstract
Midterm Report
Requirements Specification
Design Specification from J2EE & .NET team
Test Specifications
Analysis Report
User Research report
Status report
Bank One meeting Agendas
Bank One and classroom meeting minutes